

Practitioner's Docket No.: 041535-0308986
Client Reference No.: 145421US

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: YUJI KANEKO Confirmation No: UNKNOWN
Application No.: TO BE ASSIGNED Group No.: UNKNOWN
Filed: March 25, 2004 Examiner: UNKNOWN

For: CONTROL APPARATUS FOR AN INDUSTRIAL MACHINE, A METHOD OF
UPDATING A PROGRAM FOR CONTROLLING AND INDUSTRIAL MACHINE,
AND INDUSTRIAL MACHINE SYSTEM

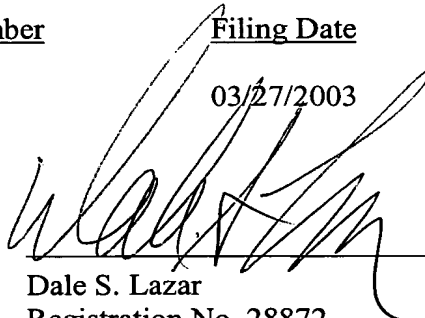
Commissioner for Patents
Mail Stop Patent Application
P.O. Box 1450
Alexandria, VA 22313-1450

SUBMISSION OF PRIORITY DOCUMENT

Attached please find the certified copy of the foreign application from which priority is
claimed for this case:

<u>Country</u>	<u>Application Number</u>	<u>Filing Date</u>
Japan	2003-088200	03/27/2003

Date: March 25, 2004
PILLSBURY WINTHROP LLP
P.O. Box 10500
McLean, VA 22102
Telephone: (703) 905-2000
Facsimile: (703) 905-2500
Customer Number: 00909


Dale S. Lazar
Registration No. 28872

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2003年 3月27日
Date of Application:

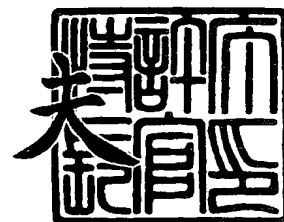
出願番号 特願2003-088200
Application Number:
[ST. 10/C]: [JP 2003-088200]

出願人 東芝機械株式会社
Applicant(s):

2004年 1月 7日

特許庁長官
Commissioner,
Japan Patent Office

今井 康



【書類名】 特許願

【整理番号】 14035301

【提出日】 平成15年 3月27日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 9/06

【発明の名称】 制御装置及びプログラムの更新方法

【請求項の数】 8

【発明者】

【住所又は居所】 静岡県沼津市大岡 2 0 6 8 の 3 東芝機械株式会社内

【氏名】 金 子 裕 司

【特許出願人】

【識別番号】 000003458

【住所又は居所】 東京都中央区銀座四丁目 2 番 1 1 号

【氏名又は名称】 東芝機械株式会社

【代理人】

【識別番号】 100075812

【弁理士】

【氏名又は名称】 吉 武 賢 次

【選任した代理人】

【識別番号】 100088889

【弁理士】

【氏名又は名称】 橘 谷 英 俊

【選任した代理人】

【識別番号】 100082991

【弁理士】

【氏名又は名称】 佐 藤 泰 和



【選任した代理人】

【識別番号】 100096921

【弁理士】

【氏名又は名称】 吉 元 弘

【選任した代理人】

【識別番号】 100103263

【弁理士】

【氏名又は名称】 川 崎 康

【手数料の表示】

【予納台帳番号】 087654

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 制御装置及びプログラムの更新方法

【特許請求の範囲】

【請求項 1】

外部機器を制御する外部機器制御プログラムを格納した、電氣的に書き換え可能な不揮発性の第 1 のメモリを備えたサブモジュールと、

前記外部機器制御プログラムを書き換えるための外部機器制御プログラム書き換え用データを格納したストレージ装置と、

前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムと、制御装置全体を制御する全体制御プログラムとを格納した第 2 のメモリを備え、通常動作時においては、前記全体制御プログラムを実行して前記サブモジュールに格納された前記外部機器制御プログラムを用いて前記サブモジュールに前記外部機器を制御させ、一方、プログラム書き換え動作時においては、前記書き換え制御プログラムを実行して、前記外部機器制御プログラムを書き換えるメインモジュールと、を備えることを特徴とする制御装置。

【請求項 2】

前記ストレージ装置は前記全体制御プログラムを書き換えるための全体制御プログラム書き換え用データをさらに格納するものであり、

前記第 2 のメモリ内の前記書き換え制御プログラムは、さらに、前記全体制御プログラム書き換え用データを用いて、前記全体制御プログラムを書き換えるものであり、

前記メインモジュールは、前記書き換え制御プログラムの実行時において、前記全体制御プログラムを書き換えることを特徴とする請求項 1 に記載の制御装置。

【請求項 3】

前記サブモジュールは、前記サブモジュールを識別する識別子を保持したサブモジュール識別機構を備え、

前記ストレージ装置は、複数の前記サブモジュール分の前記外部機器制御プロ

グラム書き換え用データを有し、

前記メインモジュールは、前記書き換え制御プログラムの実行時において、前記モジュール識別機構から前記識別子を取得し、取得した前記識別子を用いて、該当するサブモジュールに対応した前記外部機器制御プログラム書き換え用データを前記ストレージ装置から抽出し、抽出した前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換えることを特徴とする請求項 1 又は 2 に記載の制御装置。

【請求項 4】

前記サブモジュールは、前記外部機器制御プログラムのバージョン情報を保持したバージョン識別機構を備え、

前記ストレージ装置内の前記外部機器制御プログラム書き換え用データは、更新後の前記外部機器制御プログラムのバージョン情報を格納しており、

前記メインモジュールは、前記書き換え制御プログラムの実行中において、前記バージョン識別機構に保持されたバージョン情報と、前記外部機器制御プログラム書き換え用データ中におけるバージョン情報とを比較し、所定の書き換え条件を満たす場合にのみ、前記外部機器制御プログラムを書き換えることを特徴とする請求項 1 乃至 3 のいずれかに記載の制御装置。

【請求項 5】

前記第 1 のメモリは、前記外部機器制御プログラムを格納したプログラム領域と、所定のデータを格納したデータ領域とを備え、

前記ストレージ装置は、さらに、前記所定のデータを書き換えるための所定のデータ書き換え用データを有し、

前記書き換え制御プログラムは、さらに、前記所定のデータ書き換え用データを用いて、前記所定のデータを書き換えるものであり、

前記メインモジュールは、前記書き換え制御プログラムの実行中において、前記ストレージ装置における、前記外部機器制御プログラム書き換え用データ及び前記所定のデータ書き換え用データのいずれか一方あるいは両方を選択し、前記プログラム領域及び前記データ領域のいずれか一方あるいは両方に書き込むことを特徴とする請求項 1 乃至 4 のいずれかに記載の制御装置。

【請求項 6】

前記ストレージ装置内の前記外部機器制御プログラム書き換え用データと前記所定のデータ書き換え用データとは、両者を識別するインデックスデータをそれぞれ有し、

前記メインモジュールは、前記書き換え制御プログラムの実行中において、前記外部機器制御プログラム書き換え用データ及び前記所定のデータ書き換え用データのいずれか一方あるいは両方を選択し、前記インデックスデータに基づいて、前記プログラム領域及び前記データ領域のいずれか一方あるいは両方に自動的に書き込むことを特徴とする請求項 5 に記載の制御装置。

【請求項 7】

外部機器を制御する外部機器制御プログラムを格納した、電氣的に書き換え可能な不揮発性の第 1 のメモリと、

前記外部機器制御プログラムを書き換える外部機器制御プログラム書き換え用データを格納したストレージ装置と、

前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムを格納した第 2 のメモリと、

通常動作時において、前記外部機器制御プログラムを実行して、前記外部機器を制御し、プログラム書き換え動作時において、前記書き換え制御プログラムを実行して、前記外部機器制御プログラムを書き換える制御部と、

を備えることを特徴とする制御装置。

【請求項 8】

通常動作時においては、電氣的に書き換え可能な不揮発性の第 1 のメモリに格納された外部機器制御プログラムを実行して外部機器を制御する制御装置における前記外部機器制御プログラムを更新するプログラム更新方法であって、

プログラム書き換え動作時において、前記制御装置は、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムを実行することによって、前記外部機器制御プログラムを書き換えるための外部機器制御プログラム書き換え用データをストレージ装置から取得し、取得した前記外部機器制御プロ

グラム書き換え用データを用いて、前記外部機器制御プログラムを書き換えることを特徴とするプログラム更新方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ダイカストマシンや射出形成機、工作機械等の産業機械の制御を行う制御装置、及び制御装置内のプログラムを更新するプログラム更新方法に関する。

【0002】

【従来の技術】

ダイカストマシンや射出形成機、工作機械等の産業機械の制御を行う制御装置は、通常、ベース基板に挿入された、メインモジュールとサブモジュールとを基本的な構成として備える。前者のメインモジュールは各サブモジュールを管理し、一方、後者のサブモジュールは、例えば、モータコントロールモジュール、カウンタモジュール、リレーコントロールモジュール等であり、各サブモジュールに接続された外部機器を制御する。より詳しくは、各サブモジュールは、外部機器を制御するプログラムを格納したEPROMを備え、このプログラムを用いて外部機器を制御する。

【0003】

各サブモジュール内のプログラムを更新するために、従来では、プログラムが書き込まれたEPROM自体を取り替える必要があった。つまり、特別な開発ツールで作成された新たなプログラムをROMライターによってEPROMへ書き込み、このEPROMを客先まで運んで、制御装置内におけるサブモジュール上のEPROMと交換していた。

【0004】

【特許文献1】

特開平10-198571号公報

【0005】

【発明が解決しようとする課題】

しかし、これらEPROMへのプログラムの書き込み及びEPROMの取り替え作業には経験を積んだ技術者が必要となる。またEPROMの製作、交換には手間がかかってしまう。特に、各サブモジュールの配置構成は、制御装置の機種によって異なるため、1台、1台の制御装置毎に、各サブモジュール内のプログラムを管理することは非常に困難となる。また、プログラムのバージョン変更がある場合などは、更に管理が困難となる。

【0006】

プログラムの更新にかかる時間と手間を短縮するために、ネットワークを介してソフトウェアを更新する方法も考案されている（例えば、特許文献1参照）。この方法は、ネットワークを介して、更新対象となる装置を遠隔操作することにより、装置内のソフトウェアの更新を可能とする方法である。

【0007】

しかし、上述したサブモジュール内のプログラムの更新にこの方法を用いるとすると、制御装置を配置した工場内に、ネットワーク設備を導入し、このネットワーク設備に制御装置を繋げることが必要となる。これは、コストアップになるとともに、ユーザにもネットワークの知識が要求されるようになる。

【0008】

本発明は、上記問題点に鑑みてなされたものであり、プログラムの更新を容易に行うことのできる制御装置及びプログラム更新方法を提供することを目的とする。

【0009】

【課題を解決するための手段】

本発明の制御装置は、前記外部機器を制御する外部機器制御プログラムを格納した、電気的に書き換え可能な不揮発性の第1のメモリを備えたサブモジュールと、前記外部機器制御プログラムを書き換えるための外部機器制御プログラム書き換え用データを格納したストレージ装置と、前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムと、制御装置全体を制御する全体制御プログラムとを格納した第2のメモリを備え、通常動作時においては、前記全体制御プログラム

を実行して前記サブモジュールに格納された前記外部機器制御プログラムを用いて前記サブモジュールに前記外部機器を制御させ、一方、プログラム書き換え動作時においては、前記書き換え制御プログラムを実行して、前記外部機器制御プログラムを書き換えるメインモジュールと、を備えることを特徴とする。

【0010】

また、本発明の別の制御装置は、外部機器を制御する外部機器制御プログラムを格納した、電氣的に書き換え可能な不揮発性の第1のメモリと、前記外部機器制御プログラムを書き換える外部機器制御プログラム書き換え用データを格納したストレージ装置と、前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムを格納した第2のメモリと、通常動作時において、前記外部機器制御プログラムを実行して、前記外部機器を制御し、プログラム書き換え動作時において、前記書き換え制御プログラムを実行して、前記外部機器制御プログラムを書き換える制御部と、を備えることを特徴とする。

【0011】

本発明のプログラム更新方法は、通常動作時においては、電氣的に書き換え可能な不揮発性の第1のメモリに格納された外部機器制御プログラムを実行して外部機器を制御する制御装置における前記外部機器制御プログラムを更新するプログラム更新方法であって、プログラム書き換え動作時において、前記制御装置は、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムを実行することによって、前記外部機器制御プログラムを書き換える外部機器制御プログラム書き換え用データをストレージ装置から取得し、取得した前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換えることを特徴とする。

【0012】

【発明の実施の形態】

図1は、本実施の形態としての制御装置の構成を示すブロック図である。

【0013】

図1に示すように、1つのメインモジュール1と、複数のサブモジュール2（

1) ~ 2 (n) と、データストレージ装置としてのメモリカードモジュール 3 とが、それぞれデータ通信のデータバス 5 に接続されている。具体的には、これらメインモジュール 1、サブモジュール 2 (1) ~ 2 (n) 及びメモリカードモジュール 3 は図示しないコネクタを備え、これらのコネクタが、データバス 5 におけるコネクタ (図示せず) に差し込まれている。これにより、メインモジュール 1、サブモジュール 2 (1) ~ 2 (n) 及びメモリカードモジュール 3 と、データバス 5 とが電氣的に接続される。

【0014】

メインモジュール 1 は、各種の演算処理を行う CPU 10 を備えている。また、メインモジュール 1 は、3 種類のメモリ、つまり、EPROM (Erasable Programmable Read Only Memory) 11、RAM (Random Access Memory) 12、電氣的にデータの書き換え (消去及び追加) が可能な、不揮発性メモリとしてフラッシュメモリ 13 を備えている。EPROM 11 には、後述するプログラム書き込み処理を実現するプログラム (書き込み制御プログラム) が書き込まれている。一方、フラッシュメモリ 13 には、制御装置全体を適正に制御するプログラム (全体制御プログラム) が格納されている。この全体制御プログラムは、各種命令と、各種データとからなり、プログラムのバージョン番号等の属性データも含む。この全体制御プログラムは、上述のプログラム書き込み処理において書き換えの対象となる。RAM 12 は、CPU 10 が各種演算処理を行うときの作業領域として用いられる。この CPU 10 には、入出力ドライバ 14 が接続されており、この入出力ドライバ 14 には、キーボードやディスプレイ装置等の入出力端末機器 15 が繋がっている。入出力ドライバ 14 は、例えば、CPU 10 と入出力端末機器 15 の間における動作速度を調整したり、電氣的特性の違いを吸収したりする。CPU 10 は、入出力ドライバ 14 及び入出力端末機器 15 を介して、作業者と情報のやり取りを行う。入出力端末機器 15 としては、例えば、タッチパネル付きのディスプレイ装置を用いてもよい。一方、CPU 10 には、通信入出力ドライバ 6 が接続されており、この通信入出力ドライバ 6 は、データバス 5 と接続されている。通信入出力ドライバ 6 は、CPU 10 による出力信号をデータバスに適した信号に変換してデータバス 5 上に送出するとともに、データバ

ス5上を流れるデータを監視し、メインモジュール1宛のデータをデータバス5から取得して、CPU10に渡す。

【0015】

サブモジュール2(1)～2(n)も、上述のメインモジュール1と同様に、CPU16(1)～16(n)、RAM18(1)～18(n)、フラッシュメモリ19(1)～19(n)、入出力ドライバ20(1)～20(n)、通信入出力ドライバ8(1)～8(n)を備えている。入出力ドライバ20(1)～20(n)には、モータ、カウンタ、リレー等の外部機器21(1)～21(n)が接続されている。フラッシュメモリ19(1)～19(n)には、これら外部機器をコントロールするプログラム(外部機器制御プログラム)が格納されている。上述の全体制御プログラムと同様に、この外部機器制御プログラムも、各種命令と、各種データとからなり、プログラムのバージョン番号等の属性データも含む。CPU16(1)～16(n)は、この外部機器制御プログラムを実行することによって、外部機器21(1)～21(n)を制御する。また、このフラッシュメモリ19(1)～19(n)には、この外部機器制御プログラムに加えて、前述のプログラム書き込み処理時において、新たな外部機器制御プログラムをこのフラッシュメモリ19(1)～19(n)に書き込むプログラム(個別書き込みプログラム)が格納されている。また、各サブモジュール2(1)～2(n)は、各サブモジュール2(1)～2(n)に固有のID番号を保有したモジュール識別機構22(1)～22(n)を備え、このID番号は、各CPU16(1)～16(n)により読み取り可能に構成されている。具体的には、各モジュール識別機構22(1)～22(n)にはI/Oポート(図示せず)において複数の信号線が設けられ、複数の信号線はそれぞれ電源の出力電位にプルアップ、または接地される。従って、例えば、信号線が8本存在する場合は、各モジュール識別機構22(1)～22(n)に対して $2^8 (= 256)$ 種類のID番号を設定することができる。各CPU16(1)～16(n)は、このI/Oポートにおける複数の信号線の各信号線の電位を読むことにより、そのサブモジュールのID番号、つまり、そのサブモジュールの種類を認識する。なお、上述から分かるように、各サブモジュール2(1)～2(n)におけるメモリは、主とし

て2種類のメモリ、つまり、RAM18(1)～18(n)とフラッシュメモリ19(1)～19(n)とから構成され、メインモジュール1のようなEPROMは例えばコストダウンの観点から設けていない。

【0016】

メモリカードモジュール3は、上述したメインモジュール1と同様に、通信入出力ドライバ9を備える。また、メモリカードモジュール3は、メモリカードコネクタ25を備え、このメモリカードコネクタ25において、メモリカード24が挿入されている。このメモリカード24は、コンパクトフラッシュ（登録商標）やスマートメディア等のフラッシュメモリであり、汎用的にパーソナルコンピュータ、特にノート型コンピュータで用いられているものと同じものである。このメモリカード24内には、前述したプログラム書き込み処理時において、新たにインストールするプログラムデータ（全体制御プログラム及び外部機器制御プログラム）が格納されている。新たなプログラムの作成は汎用のパーソナルコンピュータで行う。作成された新たなプログラムは、この汎用のパーソナルコンピュータから、このパーソナルコンピュータのスロットに差し込まれたメモリカードに転送される。このメモリカードを、上述のメモリカードコネクタ25に差し込んで、このメモリカード内のプログラムデータを制御装置に転送する。このようにプログラムの書き換えに当たっては汎用的なメモリカードを用いるため、特別な開発ツールや媒体は必要ない。

【0017】

図2は、メモリカード24に格納されたデータ構造（ファイル）を示す図である。

【0018】

このデータ構造は、メインモジュール1に全体制御プログラムを書き込むためのメインモジュール1データと、各サブモジュール2(1)～2(n)に外部機器制御プログラムを書き込むためのサブモジュール2(1)～2(n)データとを有する。このデータ構造は、所定の基準となるアドレスから、例えば8ビット単位のブロックが、順次で並んでなるものである。以下、メインモジュール1データ及び各サブモジュール2(1)～2(n)データについてより詳細に説明す

る。

【0019】

メインモジュール1データ及び各サブモジュール2(1)～2(n)データの先頭ブロック(第1のブロック)には、モジュールの種類を表すID番号が格納される。メインモジュールの場合は、ID番号は“0”であり、サブモジュールの場合は、上述したモジュール識別機構22(1)～22(n)に設定されたID番号である。

【0020】

この第1のブロックに続くブロック(第2のブロック)には、後述する本体データ(新たなプログラム)が格納されたブロック数を示す数値が格納される。例えば、サブモジュール2(1)データの場合、本体データが格納されたブロックの数は、図2に示すように、“ $m1-4+1$ ”個であるので、この第2のブロックには“ $m1-4+1$ ”が格納される。

【0021】

この第2のブロックに続くブロック(第3のブロック)には、本体データ(新たなプログラム)のバージョン番号が格納される。バージョン番号は、例えば、最初に作成したプログラムを“0”とし、更新する毎に数が増えていく。

【0022】

以上に説明した1～3番目までのブロックのデータを、図2に示すように、インデックスデータと称する。

【0023】

この第3のブロックに続くブロック(第4のブロック)から m_x ($x=1, 2, \dots, n$)番目のブロックまでは、新たに格納するプログラム(全体制御プログラムあるいは外部機器制御プログラム)が複数のブロックに分割されて格納されている。これらのブロックに格納されたプログラムは、図2に示すように、本体データと称される。

【0024】

以上のような構成を有するメインモジュール1データ及びサブモジュール2(1)～2(n)データの内、最下位のサブモジュール2(n)データの後は、

データ構造（ファイル）の終わりを示す記号「F E N D」が格納される。

【0025】

当然ながら、プログラムの書き込み対象となるモジュール（メインモジュール 1 及びサブモジュール 2（1）～2（n））については任意に選択することができる。この場合、書き込み対象となるモジュールについてのみ、インデックスデータ及び本体データを作成すればよい。

【0026】

また、上述のように、メインモジュール 1 も、プログラムの書き換え対象となるモジュールの 1 つであるので、メインモジュール用のインデックスデータ及び本体データのみを含め、サブモジュール用のものを含めないことも可能である。

【0027】

また、各モジュールの最終データに、図 2 に示すように、チェックサムを格納する場合もある。チェックサムとは、データ内容を足し算した結果を下の桁からいくつか表示した数値である。チェックサムを格納することで、例えば、メモリカード 24 上のチェックサムの値と、フラッシュメモリに書き込んだ値のチェックサムの値とが異なっている場合は「書き込みエラー」を表示したり、またはエラーとなったモジュールデータの書き込み処理を再び行ったりすることができる。

【0028】

図 3 は、書き込み制御プログラムの動作アルゴリズムを示すフローチャートである。この書き込み制御プログラムは、上述したように、メインモジュール 1 における E P R O M 11 内に格納されており、メインモジュール 1 の C P U 10 によって解釈及び実行される。以下、このアルゴリズムに従った、プログラム書き込み処理について説明する。書き込むプログラムは、全体制御プログラム及び外部機器制御プログラムである。より詳しくは以下の通りである。

【0029】

まず、図 3 のステップ 10 に示すように、メインモジュール 1 における E P R O M 11 に格納された書き込み制御プログラムを起動する（ステップ S 10）。より詳しくは、まず、メモリカードモジュール 3 におけるコネクタ 25 に、イン

デックスデータ及び本体データの入ったメモリカード24を差し込む。この状態において、例えば、制御装置に設けられた特定のキー（図示せず）を押しながら、制御装置の電源を立ち上げる。このとき、メインモジュール1のEPROM11内格納プログラム（図示せず）が起動し、このプログラムの指示に従って、書き込み制御プログラムがCPU10に読み込まれ及び解釈される。これにより、本書き込み制御プログラムに従った処理が開始される。

【0030】

このとき、各サブモジュール2(1)～2(n)は、書き込み処理のための準備を開始する（ステップS11）。より詳しくは、各サブモジュール2(1)～2(n)におけるCPU16(1)～16(n)は、フラッシュメモリ19(1)～19(n)内に格納された個別書き込みプログラムをRAM18(1)～18(n)上にロードする。各サブモジュール2(1)～2(n)は、個別書き込みプログラムをRAM18(1)～18(n)上にロードしたら、準備の完了を示す準備完了フラグをデータバス5上に送出する。CPU10は、データバス5上に送出された準備完了フラグを受信するまで待機する（ステップS11のNo）。

【0031】

CPU10は、全てのサブモジュール2(1)～2(n)から準備完了フラグを受信したら（ステップS11のYes）、各サブモジュール2(1)～2(n)から順次、ID番号を読み出して、制御装置におけるサブモジュールの配置構成を把握する（ステップS12）。より詳しくは、各サブモジュール2(1)～2(n)におけるCPU16(1)～16(n)が、CPU10からID番号取得指令を受けて、ID番号設定部22(1)～22(n)に保有されたID番号を取得し、取得したID番号をCPU10に送出する。ID番号を取得したCPU10は、取得したID番号に基づき、サブモジュールの配置構成を把握する。

【0032】

CPU10は、サブモジュールの構成を把握したら、次に、メモリカード24内の先頭ブロック（第1のブロック）（図2参照）内のデータを読み込む（ステップS13）。

【0033】

CPU10は、読み込んだ先頭ブロック内のデータがデータ終了記号“FEND”（図2参照）であるかどうかを判断する（ステップS14）。

【0034】

CPU10は、読み込んだ先頭ブロック内のデータがデータ終了記号“FEND”であると判断した場合は（ステップS14のYes）、プログラム書き込み処理を終了する。

【0035】

一方、CPU10は、読み込んだ先頭のブロック内のデータがデータ終了記号“FEND”（図2参照）でないと判断した場合は（ステップS14のNo）、つまり、その先頭ブロック内のデータはID番号であると判断した場合は（図2参照）、そのID番号を有するモジュール（メインモジュール1あるいはサブモジュール2（1）～2（n））が制御装置内に存在するかどうかを判断する（ステップS15）。具体的には、CPU10は、上のステップS12で取得したID番号の中に、この先頭ブロック内のID番号と同じものが存在するかどうかを判断する（ステップS15）。

【0036】

CPU10は、ステップS12で取得したID番号の中に、先頭ブロック内のID番号と同じものが存在しないと判断した場合は（ステップS15のNo）、以下のステップS16、S17を飛ばして、続くステップS18に進む。

【0037】

一方、CPU10は、ステップS12で取得したID番号の中に、先頭ブロック内のID番号と同じものが存在すると判断した場合は（ステップS15のYes）、この先頭ブロックに続くブロック（第2のブロック）内のデータ、つまり、データ数（ブロック数）（図2参照）を取得し、RAM12に格納する。さらに、CPU10は、この第2のブロックに続くブロック（第3のブロック）内のデータ、つまり、バージョン番号（図2参照）を取得し、書き換え対象となるプログラムのバージョン番号と比較する（ステップS16）。より詳しくは、書き換え対象がメインモジュール1である場合は、CPU10は、第3のブロック内

のバージョン番号を、フラッシュメモリ 13 内の全体制御プログラムのバージョン番号（前述したように全体制御プログラム中に含まれる）と比較する。一方、書き換え対象が各サブモジュール 2（1）～2（n）である場合は、CPU 10 は、第 3 のブロック内のバージョン番号を、書き込み対象となるサブモジュール 2 に設けられたフラッシュメモリ 19 内の外部機器制御プログラムのバージョン番号（前述したように外部機器制御プログラム中に含まれる）と比較する。より詳しくは、CPU 10 は、書き込み対象となるサブモジュール 2 における各 CPU 16 にバージョン番号取得指令を出し、このバージョン番号取得指令を受けた CPU 16 が、フラッシュメモリ 19 内の外部機器制御プログラムからバージョン番号を取得して、CPU 10 に送出する。CPU 10 は、第 3 のブロック内のバージョン番号と、受信したバージョン番号とを比較する。

【0038】

比較の結果、CPU 10 が、第 3 のブロック内のバージョン番号が、書き換え対象となるプログラムのバージョン番号以下であると判断した場合は（ステップ S 16 の No）、以下のステップ S 17 を飛ばして、続くステップ S 18 に進む。

【0039】

一方、比較の結果、CPU 10 が、第 3 のブロック内のバージョン番号が、書き換え対象となるプログラムのバージョン番号よりも大きいと判断した場合は（つまり新たに格納するプログラムが最新のものである場合は（ステップ S 16 の Yes）以下の処理を行う。まず、書き換え対象がメインモジュール 1 である場合は、CPU 10 は、フラッシュメモリ 13 内のデータを全て消去し、1 ユニット（1 モジュール）分の本体データ（図 2 参照）をメモリカード 24 から読み込み、読み込んだ本体データをフラッシュメモリ 13 内に書き込む（ステップ S 17）。1 ユニット分の本体データの読み込みは、上のステップ S 16 において取得したブロック数分のデータを、上述のバージョン番号が格納された第 3 のブロックの次のブロック（第 4 のブロック）から読み込むことで行う（図 2 参照）。一方、書き換え対象が各サブモジュール 2（1）～2（n）である場合は、CPU 10 は、該当するサブモジュール 2 の CPU 16 に対してプログラム書き換え

の指示を出す（ステップS17）。この指示を受信したCPU16は、RAM18上にロードされた個別書き込みプログラムを実行して以下の処理を行う（ステップS17）。即ち、まず、CPU16は、フラッシュメモリ19内のデータ（外部機器制御プログラム及び個別書き込みプログラム）を全て消去する（ステップS17）。次いで、CPU10が、メモリカード24内の本体データ（図2参照）をメモリカード24から読み込み、読み込んだ本体データ（新たな外部機器制御プログラム）を通信入出力ドライバ8を介してCPU16へ渡す。CPU16は、受け取った本体データをフラッシュメモリ19に書き込む（ステップS17）。この後、CPU16は、RAM18上にロードした個別書き込みプログラムをフラッシュメモリ19内に書き込む（ステップS17）。なお、個別書き込みプログラムによる書き込みは、予めこの個別書き込みプログラムを上述の本体データとしてメモリカード24内に格納しておき、上述の新たな外部機器制御プログラムと共にフラッシュメモリ19に書き込むようにしてよい。

【0040】

CPU10は、フラッシュメモリ内へのプログラムの書き込みが終わったら、メモリカード24のアクセスポインタを次のモジュールデータ（メインモジュールデータあるいはサブモジュールデータ）（図2参照）の先頭にセットし（ステップ18）、上述のステップ13～S17を、データ終了記号“FEND”を読み込むまで（ステップS14のYes）、繰り返す。

【0041】

CPU10は、データ終了記号“FEND”を読み込んだら（ステップS14のYes）、書き込み制御プログラムを終了する。この後、制御装置を再起動させると、通常の動作モードとなり、新たにフラッシュメモリ13に書き込まれた全体制御プログラム、及び新たに各フラッシュメモリ19（1）～19（n）に書き込まれた外部機器制御プログラムが実行される。

【0042】

上述したプログラムの書き込み処理において新たな外部機器制御プログラムを書き込む際には、フラッシュメモリ19内のデータを全消去するため、個別書き込みプログラムを一旦フラッシュメモリ19からRAM18上にロードして実行

した。しかし、この個別書き込みプログラムを別のメモリ、例えばE P R O Mに格納し、このE P R O Mを各サブモジュール 2 (1) ~ 2 (n) に設けることで、このE P R O M内の個別書き込みプログラムをR A M 1 8上にロードすることなく、E P R O M上で直接、実行してもよい。これによれば、R A M 1 8上に個別書き込みプログラムをロードしたり、フラッシュメモリ 1 9に個別書き込みプログラムを書き込んだりする必要はないので、プログラムの書き込み処理を効率化できる。

【0043】

また、上述のように、プログラムの書き込み処理においては、フラッシュメモリ内のデータを一旦全部消去して、新たにプログラムをフラッシュメモリに書き込むこととした。しかし、フラッシュメモリ内のプログラムの一部を書き換えたり、新たに差分プログラムを追加したりすることにより、プログラムを更新するようにしてもよい。この場合は、メモリカード 2 4内には、一部書き換え用のデータや追加分の差分データを用意すればよい。

【0044】

また、当然ながら、1つの制御装置内に同じ種類のサブモジュールが複数装備されている場合もあり、そのような場合にも、すべてのサブモジュールについて、上述のプログラム書き込み処理を行うことができる。

【0045】

ところで、制御装置は、例えば各サブモジュール 2 に接続された外部機器の表示部に、外部機器の操作の仕方を示すヘルプ画面としての操作画面データを表示して、作業者に対し操作のガイドする場合がある。この場合、例えば、この操作画面データは、上述したフラッシュメモリ 1 9内に格納される。このフラッシュメモリ 1 9内には、前述したように、個別書き込みプログラムも格納されている。このような状態において、フラッシュメモリ 1 9内に格納された個別書き込みプログラム及び操作画面データのうち、いずれか一方のみを更新したい場合もある。つまり、個別書き込みプログラム及び操作画面データをそれぞれ独立して更新したい場合がある。このためには、上述のサブモジュール 2 (1) ~ 2 (n) のうちの任意のサブモジュール 2 (m) (m=1、2、3・・・n) を示す図 4

から分かるように、例えば、フラッシュメモリ 19 (m) の領域を、個別書き込みプログラム及び外部機器制御プログラムを格納するプログラム領域 19 a と、画像データを格納するデータ領域 19 b とに分割する。そして、上述のプログラム書き込み処理時において、プログラム領域 19 a 内のプログラムとデータ領域 19 b 内の画像データのいずれか一方あるいは両方を書き換えるようにすればよい。これを実現するためのデータ構造を図 5 に示す。

【0046】

図 5 に示すように、このデータ構造においては、ID 番号の格納されたブロックの次のブロックに、書き換え対象がプログラムあるいは画像データであることを示す記号（データ種類）が格納される。この例では、プログラムの場合は“p”、画像データの場合は“d”が格納されている。上述した書き替え制御プログラムを実行する CPU 10 は、このデータ種類“p”あるいは“d”が格納されたブロックを読み込み、このブロック内のデータが“p”（プログラム）であると判断した場合は、本体データ（図 5 参照）をフラッシュメモリ 2 (m) のプログラム領域 19 a に書き込む。一方、CPU 10 は、ブロック内のデータが、“d”（画像データ）であると判断した場合は、本体データ（画像データ）をデータ領域 19 b に書き込む。これにより、プログラムと画像データをそれぞれ独立して自動的に書き換えることができる。

【0047】

図 4 及び図 5 に示す書き換え例では、サブモジュールを対象としたが、メインモジュールにおいても、同様に、プログラムと画像データとをそれぞれ独立に更新することができる。また、データ領域 19 b には画像データを格納することとしたが、その他のデータ、例えば、プログラムのバージョン番号なども、プログラム領域 19 a でなく、このデータ領域 19 b に格納してもよい。

【0048】

以上のように本実施の形態によれば、容易にプログラムや画像データ等のデータを更新することができる。

【0049】

即ち、サブモジュールの配置構成の違いにより多種の制御装置があるが、各制

御装置のプログラム及びデータを更新する場合でも、各制御装置分の更新用のデータを1つにまとめ、1つにまとめられた更新用のデータを各制御装置に用いることができる。これは、1つにまとめられたデータのうち、ある制御装置にとって必要な更新用データのみが、前述したID番号に基づき抽出され、その制御装置に書き込まれるからである。

【0050】

また、プログラムの書き込み処理時において、プログラムのバージョン番号も自動的にチェックし、最新のバージョン番号の方を優先するようにしたので、プログラムのバージョン管理も簡単になる。

【0051】

また、更新用のデータを格納する媒体として汎用的なメモ리카ードを用いるようにしたので、特別な装置の開発は必要ない。これにより、例えば、制御装置の配置された現場と、プログラム等を開発する部署が離れていても、迅速に制御装置内のプログラム及びデータを更新することができる。つまり、書き込み用のデータを、制御装置の近くにあるパーソナルコンピュータに開発部署からネットワークを介して送出し、この書き込み用のデータをこのパーソナルコンピュータで受信する。次いで、この書き込み用のデータをこのパーソナルコンピュータから、このパーソナルコンピュータに差し込まれたメモ리카ードに転送し、書き込み用のデータが格納されたメモ리카ードをパーソナルコンピュータから抜いて、制御装置のコネクタに差し込む。この後、上述した本実施の形態を用いて、制御装置内に格納された各種のプログラム及びデータを更新する。これによれば、従来のように、新たなプログラムを格納したEPROMを現場に運んで交換するという作業は必要ないので、素早くプログラム及びデータの更新をすることができる。

【0052】

以上に述べた本実施の形態では、ストレージ装置としてメモ리카ードを用いたが、その他、例えば、ハードディスクや、磁気ディスク、光ディスクを用いることもできる。

【0053】

また、本実施の形態では、メモリカードを制御装置本体に直接差し込むことによりメモリカード内のデータを読み出したが、このメモリカード内のデータをインターネット等のネットワーク越しに読み出すようにしてもよい。

【0054】

また、本実施の形態では、プログラムの更新対象として、全体制御プログラムと外部機器制御プログラムを選択したが、この他に、例えば、書き込み制御プログラムや、個別書き込みプログラム自体も更新の対象とすることができる。書き込み制御プログラムの更新では、例えば、書き換え対象となるプログラムを上述のE P R O Mではなく、E E P R O M (Electrically Erasable Programmable Read-Only Memory) やフラッシュメモリに格納し、書き込み制御プログラムの実行時において、この書き込み制御プログラムをR A M上にロードして実行する。そして、新たな書き込み制御プログラムをストレージ装置から読み出し、上述のE E P R O Mやフラッシュメモリに書き込む。一方、個別書き込みプログラムの更新の場合も、同様に、更新時において、個別書き込みプログラムをR A M上にロードして実行し、ストレージ装置から読み出した新たな個別書き込みプログラムをフラッシュメモリに書き込む。

【0055】

また、本実施の形態では、各モジュールに対する書き込み用のプログラムを1つのファイルにまとめ、この1つのファイルを用いて、各モジュールに対し一括してプログラムを書き込むようにした。しかし、書き込み用のプログラムを格納したファイルをモジュールごとに作成し、マニュアルモードなどにより、書き換えるモジュールとそれに対応したファイルを選択するようにすることで、各モジュールに対してそれぞれ個別のファイルを用いてプログラムを書き込むようにしてもよい。

【0056】

また、本実施の形態では、書き込むプログラムのバージョンが、書き換え対象となるプログラムのバージョンよりも古いときは、プログラムの書き換えを行わないようにしたが、書き換え制御プログラムのオプションとして、プログラムのバージョンにかかわらず（例えば古いバージョンのものであっても）書き込み可

能とするモードを設けてもよい。これは、バージョンアップされたプログラムにおいて後から不具合な点などが発見された場合に、前のバージョンのプログラムに戻す処理を可能とするためである。

【0057】

【発明の効果】

本発明によれば、ストレージ装置から読み出した更新用のプログラムを用いて、電氣的に書き換え可能な不揮発性のメモリ内に格納された外部機器制御プログラムを書き換えるようにしたので、外部機器制御プログラムを容易に更新することができる。

【図面の簡単な説明】

【図1】

本実施の形態としての制御装置の構成を示すブロック図である。

【図2】

フラッシュメモリ内のプログラムを更新するためのデータ構造を示す図である。

【図3】

書き込み制御プログラムの動作アルゴリズムを示すフローチャートである。

【図4】

サブモジュールにおけるフラッシュメモリの領域をプログラム領域とデータ領域とに分けた状態を示す図である。

【図5】

プログラムと画像データとをそれぞれ独立にデータ更新するためのデータ構造を示す図である。

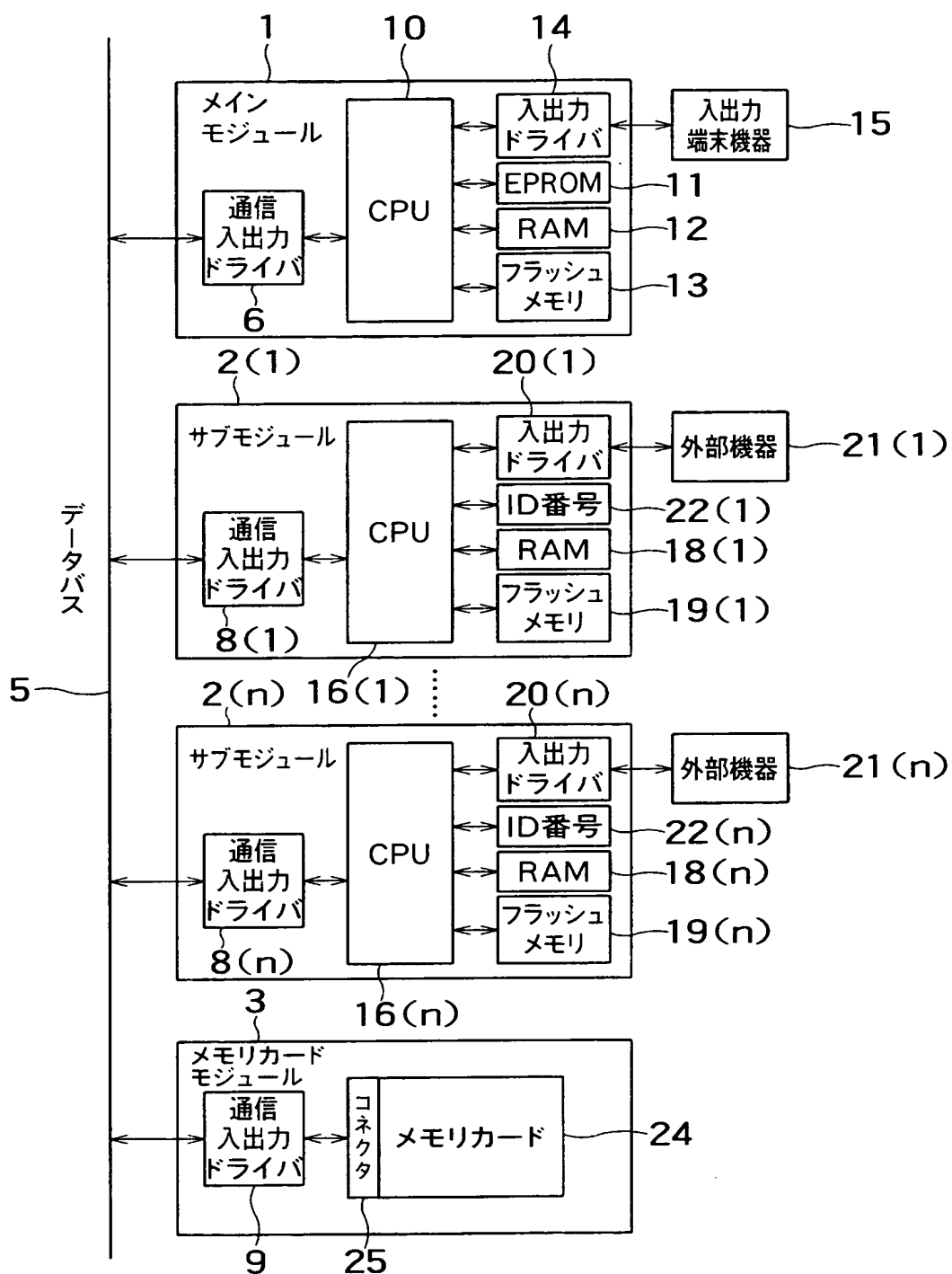
【符号の説明】

- 1 メインモジュール
- 2 サブモジュール
- 3 メモリカードモジュール
- 5 データバス
- 6、8、9 通信入出力ドライバ

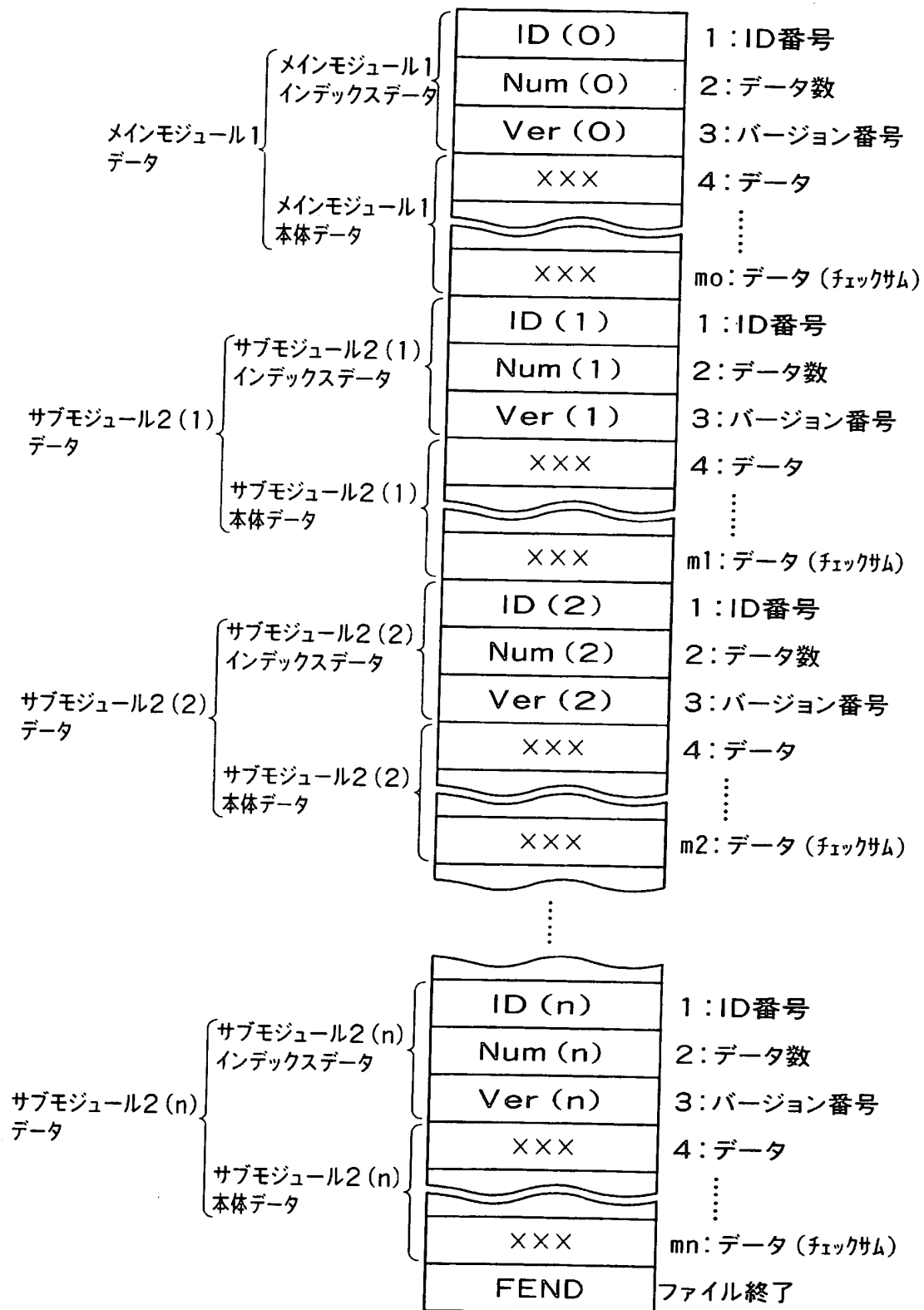
- 1 0、1 6 C P U
- 1 1 E P R O M
- 1 2、1 8 R A M
- 1 3、1 9 フラッシュメモリ
- 1 4、2 0 入出力ドライバ
- 1 5 入出力端末機器
- 1 9 a プログラム領域
- 1 9 b データ領域
- 2 1 外部機器
- 2 2 モジュール識別機構 (I D 番号)
- 2 4 メモリカード
- 2 5 コネクタ

【書類名】 図面

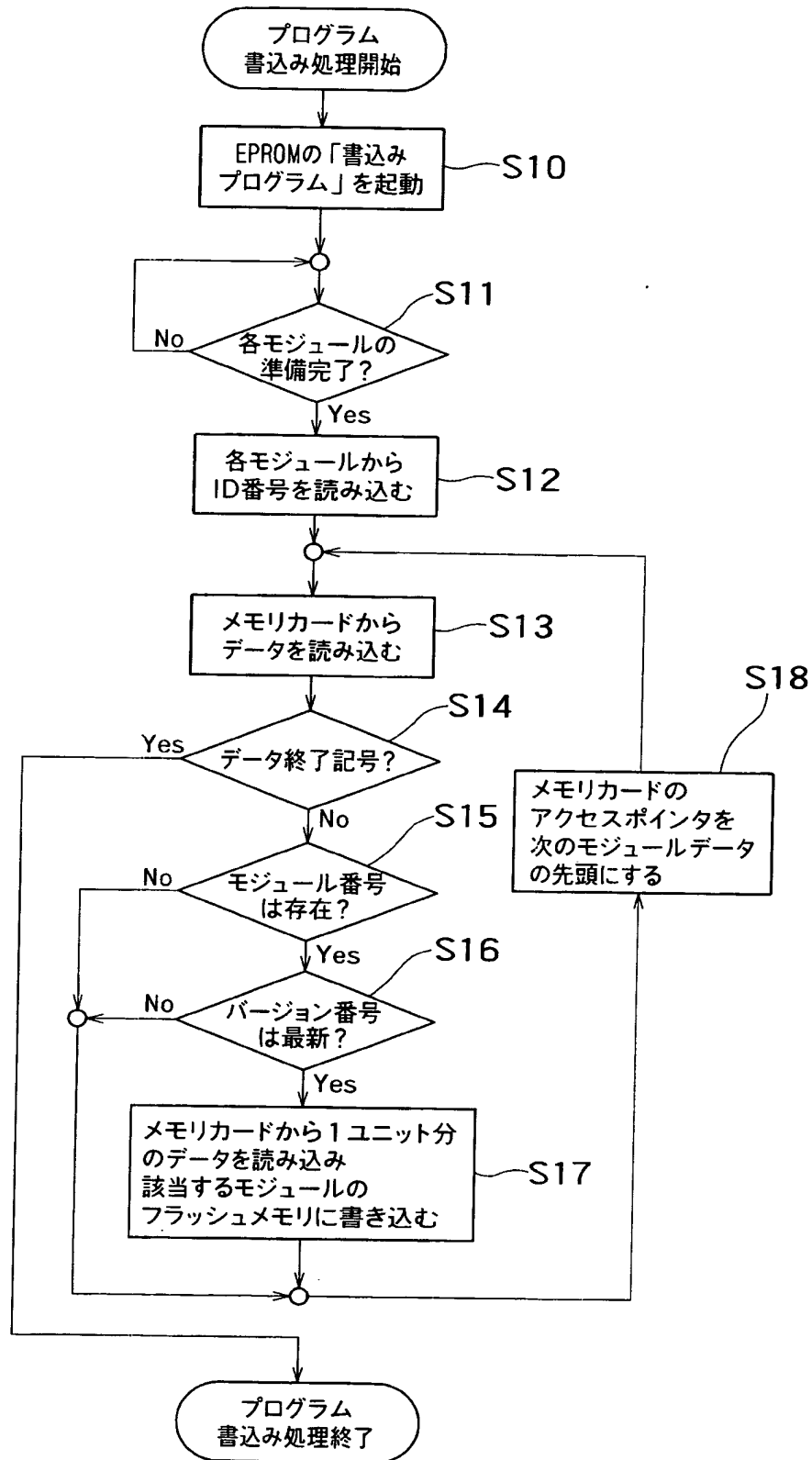
【図 1】



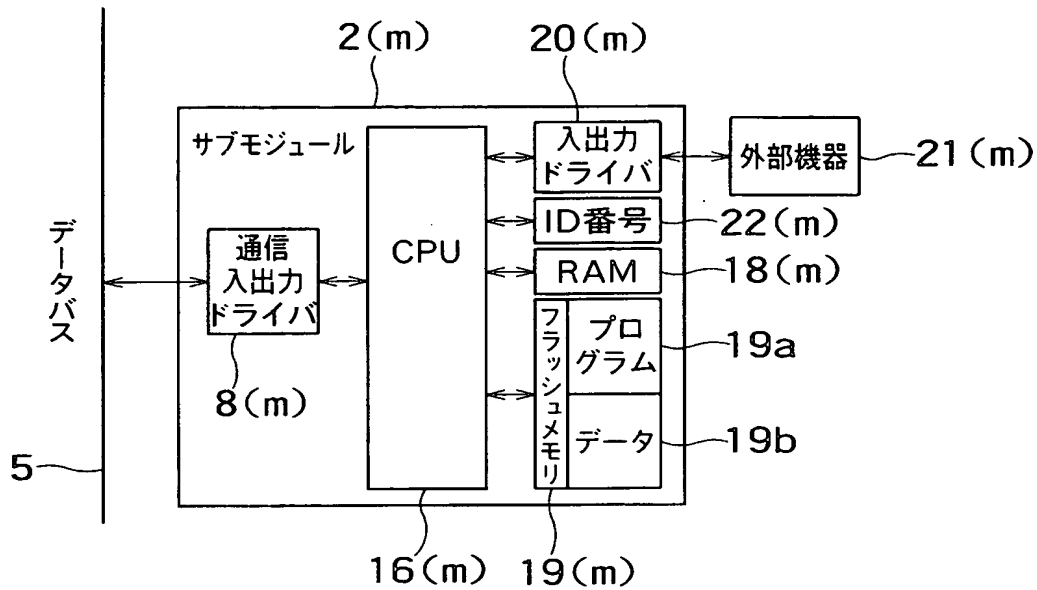
【図 2】



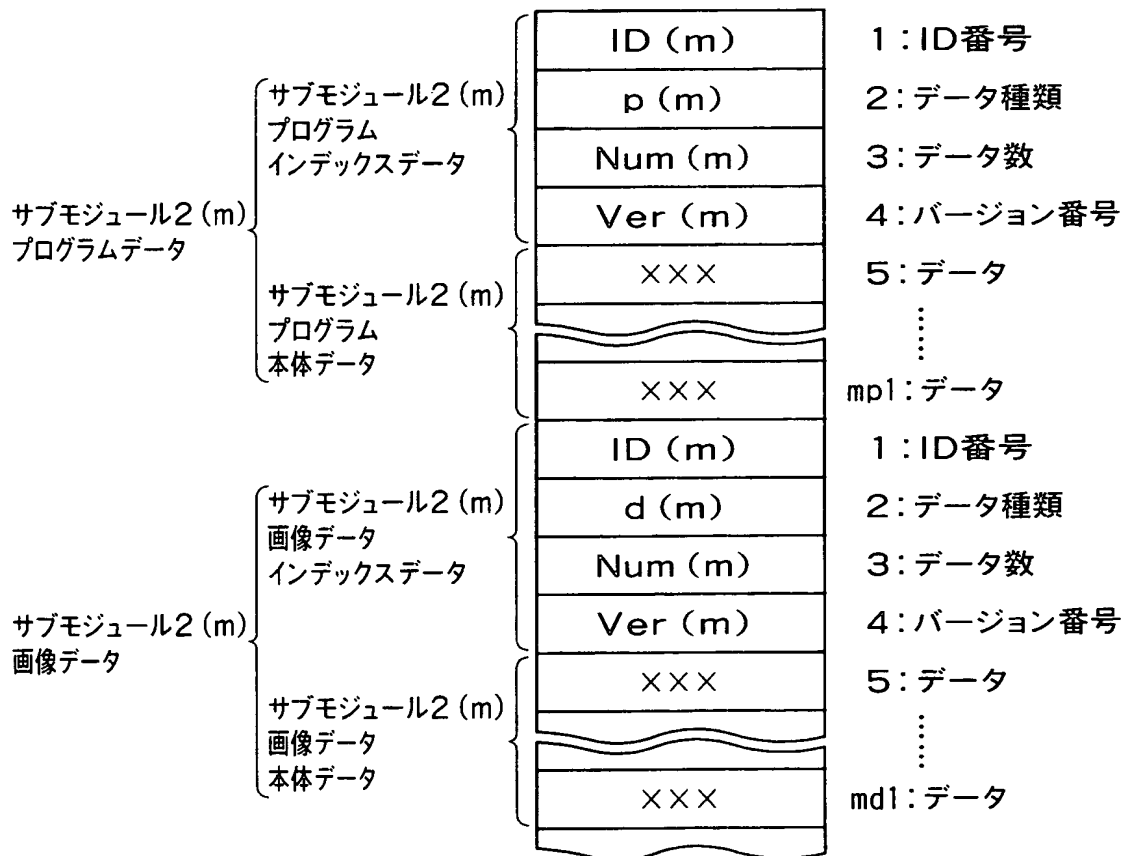
【図 3】



【圖 4】



【図 5】



【書類名】 要約書

【要約】

【課題】 工作機械等の産業機械を制御する制御装置におけるプログラムやデータの更新を容易に行う。

【解決手段】 通常動作時には、電氣的に書き換え可能な不揮発性の第1のメモリに格納された外部機器制御プログラムを実行して外部機器を制御する制御装置において、プログラム書き換え動作時には、前記制御装置は、前記外部機器制御プログラムを書き換える処理を実現する書き換え制御プログラムを実行することによって、前記外部機器制御プログラムを書き換える外部機器制御プログラム書き換え用データをストレージ装置から取得し、取得した前記外部機器制御プログラム書き換え用データを用いて、前記外部機器制御プログラムを書き換える。

【選択図】 図1

特願 2 0 0 3 - 0 8 8 2 0 0

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 3 4 5 8]

1. 変更年月日

1 9 9 0 年 8 月 2 7 日

[変更理由]

新規登録

住 所

東京都中央区銀座 4 丁目 2 番 1 1 号

氏 名

東芝機械株式会社

2. 変更年月日

2 0 0 3 年 5 月 2 6 日

[変更理由]

住所変更

住 所

東京都中央区銀座 4 丁目 2 番 1 1 号

氏 名

東芝機械株式会社